



FPT POLYTECHNIC



www.poly.edu.vn

LẬP TRÌNH ANDROID 1

DATA STORAGE

- Biết cách sử dụng Internal Storage
- Biết cách sử dụng Shared Preferences



LẬP TRÌNH ANDROID 1

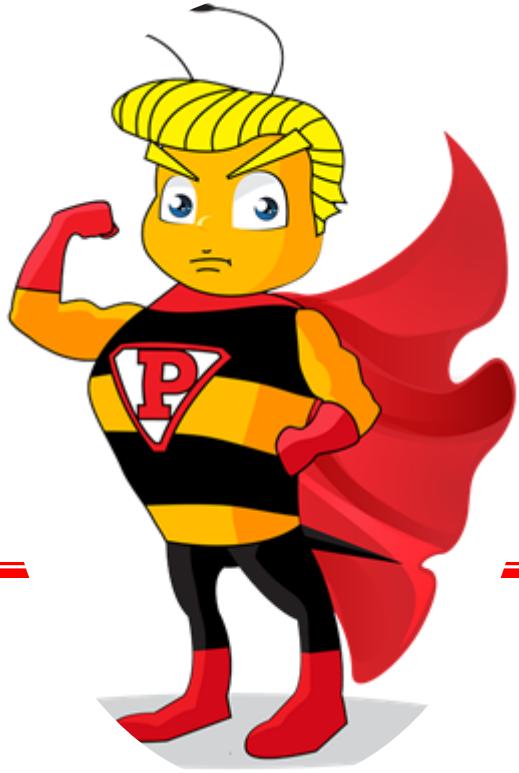
BÀI 8.1: INTERNAL STORAGE

MỤC TIÊU

- BIẾT CÁCH SỬ DỤNG INTERNAL STORAGE
- BIẾT CÁCH SỬ DỤNG SHARED PREFERENCES

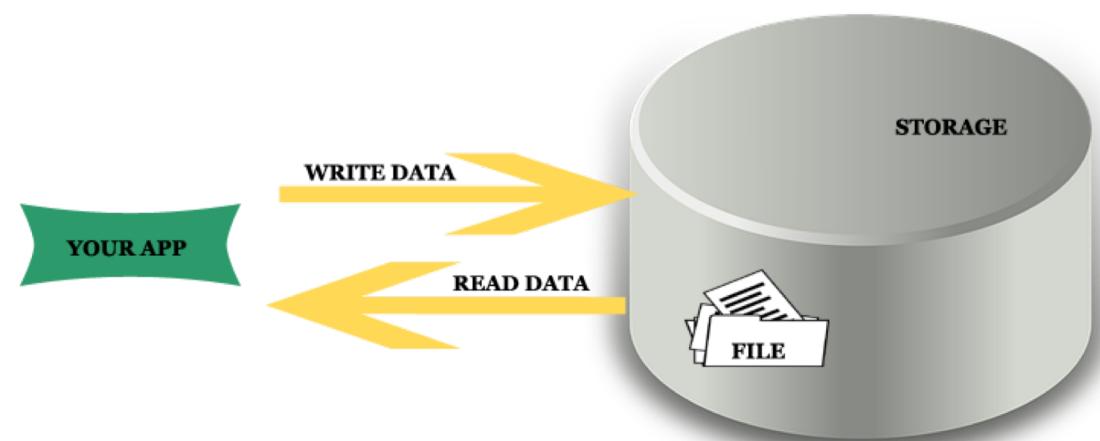


- ❑ Khi phát triển ứng dụng, tuỳ thuộc vào các đặc điểm và yêu cầu của từng ứng dụng thì việc duy trì thông tin về trạng thái, cấu hình cho mỗi lần ứng dụng chạy bằng cách lưu trữ dữ liệu cục bộ trên thiết bị di động là cần thiết.
- ❑ Android cung cấp một số tùy chọn hỗ trợ cho việc lưu trữ dữ liệu trong một ứng dụng như sau:
 - ❖ Shared Preferences
 - ❖ Internal/External Storage
 - ❖ SQLite Database
 - ❖ Content Provider



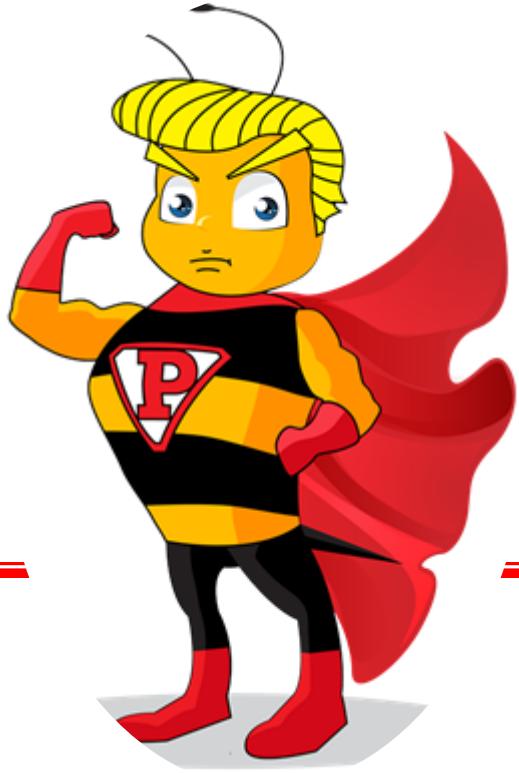
INTERNAL STORAGE

- ❑ Để đơn giản hóa việc đọc và ghi luồng dữ liệu tập tin được lưu ở bộ nhớ trong, Android cung cấp các phương thức **openFileInput / openFileOutput** tương ứng cho 2 luồng đọc và ghi từ context của phiên làm việc hiện tại của ứng dụng.
- ❑ Nếu tập tin có tên là giá trị được truyền làm đối số của hàm **FileOutputStream()** không tồn tại (tập tin chưa được tạo ra và lưu lại), hệ thống sẽ tự động tạo tập tin và ghi xuống bộ nhớ. Nếu đã có sẵn thì hệ thống mặc định sẽ chép đè nội dung mới lên nội dung cũ của tập tin.



❑ Một số điểm quan trọng trong việc lưu trữ trong bộ nhớ trong

- ❖ Dữ liệu trong bộ nhớ được phép đọc và ghi vào tập tin.
- ❖ Các tập tin này chỉ được truy cập bởi chính ứng dụng đó, không cho phép truy cập bởi ứng dụng khác.
- ❖ Các tập tin này vẫn tồn tại khi ứng dụng vẫn tồn tại, các tập tin sẽ bị xóa tự động khi ứng dụng uninstall.
- ❖ Các tập tin này lưu trữ trong thư mục data/data theo sau là tên package của ứng dụng.
- ❖ Có thể phân quyền cho các ứng dụng khác truy cập các tập tin này.
- ❖ Để phân quyền dữ liệu chúng ta có thể dùng chế độ MODE_PRIVATE



INTERNAL STORAGE LUU DỮ LIỆU VÀO FILE CACHCE

Các Mode lưu trữ ở bộ nhớ trong

- ❖ **MODE_PRIVATE** – Đây là chế độ mặc định. *MODE_PRIVATE* được sử dụng để thiết lập chế độ riêng tư, không chấp nhận ứng dụng từ bên ngoài.
- ❖ **MODE_APPEND** – Dữ liệu được thêm vào tập tin đã tồn tại



☐ File cache được lưu trong **data/data/<package_app_name>/cache**

> com.breel.wallpapers18	drwxrwx-->	2023-04-02 11:0	4 KB
▼ com.dinhnt.firstproject	drwxrwx-->	2023-04-02 11:0	4 KB
> .agent-logs	drwx-----	2023-04-02 20:0	4 KB
▼ cache	drwxrws-->	2023-04-02 20:0	4 KB
data.cache	-rw-----	2023-04-02 20:0	15 B
> code_cache	drwxrws-->	2023-04-02 20:0	4 KB
> files	drwxrwx-->	2023-04-02 20:0	4 KB
> com.google.android.apps	drwxrwx-->	2023-04-02 11:0	4 KB
.....

☐ Lưu dữ liệu vào file Cache

```
String data = "FPT Polytechnic";  
File pathCacheDir = getCacheDir();  
File file = new File(pathCacheDir, "data.cache");  
try {  
    file.createNewFile();  
    FileOutputStream outputStream = new FileOutputStream(file.getAbsolutePath());  
    outputStream.write(data.getBytes());  
    outputStream.close();  
} catch (IOException e) {  
    throw new RuntimeException(e);  
}
```

❑ Đọc dữ liệu từ file Cache

```
try {  
    File pathCacheDir = getCacheDir();  
    File file = new File(pathCacheDir, "data.cache");  
    Scanner scanner = new Scanner(file);  
    StringBuilder data = new StringBuilder();  
    while (scanner.hasNext()) {  
        data.append(scanner.nextLine());  
    }  
    scanner.close();  
    Toast.makeText(this, data.toString(), Toast.LENGTH_SHORT).show();  
} catch (Exception e) {  
    e.printStackTrace();  
}
```

❑ Lấy toàn bộ file Cache và xóa chúng

```
File patchCacheDir = getCacheDir();
File[] listCache = patchCacheDir.listFiles();
if (listCache != null) {
    for (File f : listCache) {
        f.delete();
    }
}
```



INTERNAL STORAGE LUU DỮ LIỆU VÀO FILE TEXT (TXT)

- ❑ File text (*txt) được lưu trong **data/data/<package_app_name>/files**

>	com.breel.wallpapers18	drwxrwx-->	2023-04-02 11:0	4 KB
	com.dinhnt.firstproject	drwxrwx-->	2023-04-02 11:0	4 KB
	cache	drwxrws-->	2023-04-02 11:0	4 KB
	code_cache	drwxrws-->	2023-04-02 20:0	4 KB
	files	drwxrwx-->	2023-04-02 20:0	4 KB
	data.txt	-rw-rw----	2023-04-02 20:0	14 B
>	com.google.android.apps	drwxrwx-->	2023-04-02 11:0	4 KB

☐ Lưu dữ liệu vào file text

```
String userName = "dinhnt ";
String passWord = "abc@123";

try {
    FileOutputStream fileOutputStream = openFileOutput("data.txt", Context.MODE_PRIVATE);
    fileOutputStream.write(userName.getBytes());
    fileOutputStream.write(passWord.getBytes());
    fileOutputStream.close();
} catch (IOException e) {
    throw new RuntimeException(e);
}
```

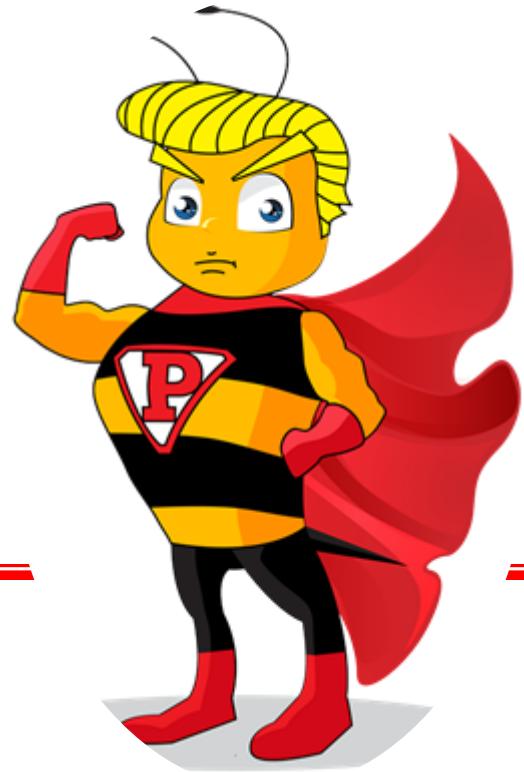
❑ Đọc dữ liệu từ file text

```
try {  
    FileInputStream fileInputStream = openFileInput("data.txt");  
    int read = -1;  
    StringBuilder buffer = new StringBuilder();  
    while((read =fileInputStream.read())!= -1){  
        buffer.append((char)read);  
    }  
    String name = buffer.substring(0, buffer.indexOf(" "));  
    String pass = buffer.substring(buffer.indexOf(" ") + 1);  
    Toast.makeText(this, name + " - " + pass, Toast.LENGTH_SHORT).show();  
} catch (IOException e) {  
    throw new RuntimeException(e);  
}
```



LẬP TRÌNH ANDROID 1

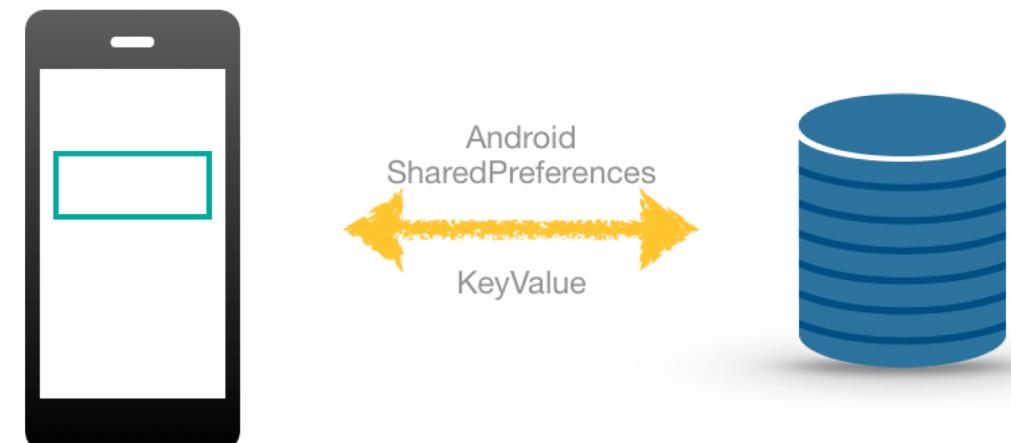
BÀI 8.2: SHARED PREFERENCES



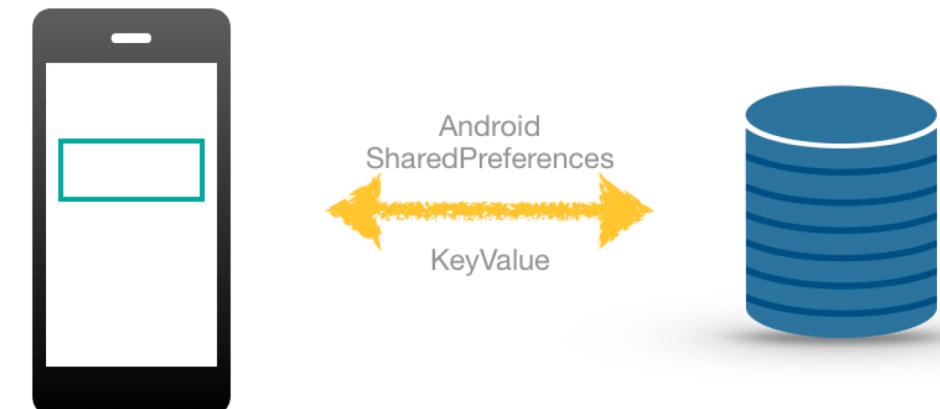
SHARED PREFERENCES

❑ **Shared Preferences** là đối tượng Android cung cấp cho việc lưu trữ và truy xuất các dữ liệu có kiểu cơ bản như *boolean*, *string*, *float*, *long*, và *integer* trong lập trình các ứng dụng đơn giản.

❖ **Ví dụ:** Ứng dụng của bạn cho phép người dùng cài đặt các tùy chọn liên quan đến màu chữ hay màu nền của các màn hình. Trong trường hợp này, bạn có thể sử dụng đối tượng Shared Preferences để lưu lại các thông tin tùy chọn người dùng thiết lập theo từng cặp key/value và truy xuất sử dụng cho mỗi lần mở lại ứng dụng.



- ❑ **Shared Preference** sẽ được lưu lại thông qua việc sử dụng chỉ định khóa cho từng giá trị dữ liệu (theo từng cặp **key/value**), những giá trị **key.value** này sẽ được tự động ghi vào một tập tin XML được chứa bên trong thư mục của ứng dụng. Những cặp giá trị này sẽ tồn tại và sẵn sàng cho việc sử dụng (đọc và cập nhật) trong suốt phiên làm việc của ứng dụng và được chia sẻ bên trong các thành phần của ứng dụng (như *activity*, *service*,...). Các giá trị Shared Preference này không thể truy xuất và sử dụng từ những ứng dụng khác.



- ❑ File SharedPreferences được lưu trong đường dẫn:

data/data/<package_app_name>/shared_prefs

▼	com.dinhnt.firstproject	drwxrwx-->	2023-04-02 11:0	4 KB
>	.agent-logs	drwx-----	2023-04-02 20:0	4 KB
>	cache	drwxrws-->	2023-04-02 20:0	4 KB
>	code_cache	drwxrws-->	2023-04-02 20:0	4 KB
>	files	drwxrwx-->	2023-04-02 20:0	4 KB
▼	shared_prefs	drwxrwx-->	2023-04-02 20:0	4 KB
	data.xml	-rw-rw----	2023-04-02 20:0	157 B
>	com.google.android.apps	drwxrwx-->	2023-04-02 11:0	4 KB

❑ Có 2 cách lưu trữ dữ liệu trong Shared Preferences

❖ Cách 1: Sử dụng Activity Preferences

Chúng ta phải gọi **getPreferences (int mode)** có trong lớp Activity.

Chỉ sử dụng khi Preferences file là cần thiết trong Activity

Nó không yêu cầu tên vì nó sẽ là tập tin ưu tiên duy nhất cho Activity.

Thông thường sẽ không sử dụng cách này

❑ Có 2 cách lưu trữ dữ liệu trong Shared Preferences

❖ Cách 2: Sử dụng Custom Preferences

Sử dụng phương thức **getSharedPreferences(String name, int mode)**
cho Custom Preferences

Được sử dụng trong các trường hợp khi yêu cầu nhiều tập tin Preferences
trong Activity

Tên của tập tin Preferences được truyền cho tham số đầu tiên

❑ Có 4 loại Mode trong Shared Preference

- ❖ **MODE_PRIVATE** - tạo file (hoặc thay thế file nếu đã có) và thiết lập file này ở chế độ **RIÊNG TƯ** đối với các ứng dụng khác có trong điện thoại
- ❖ **MODE_APPEND** - nối thêm dữ liệu vào file đã có trước đó
- ❖ **MODE_WORLD_READABLE** - có thể đọc file từ ứng dụng khác
- ❖ **MODE_WORLD_WIRTEABLE** - có thể đọc/ghi từ ứng dụng khác

❑ Lưu dữ liệu vào file Shared Preferences

```
String fullName = "Nguyễn Văn A";
int old = 27;

SharedPreferences sharedPreferences = getSharedPreferences("data", MODE_PRIVATE);
SharedPreferences.Editor editor = sharedPreferences.edit();

editor.putString("fullname", fullName);
editor.putInt("old", old);

editor.apply();
```

❑ Đọc dữ liệu từ file Shared Preferences

```
SharedPreferences sharedPreferences = getSharedPreferences("data", MODE_PRIVATE);
String name = sharedPreferences.getString("fullname", "");
int old = sharedPreferences.getInt("old", 0);
Toast.makeText(this, name + " - " + old, Toast.LENGTH_SHORT).show();
```

Xóa dữ liệu từ file Shared Preferences

```
SharedPreferences sharedPreferences = getSharedPreferences("data", MODE_PRIVATE);
SharedPreferences.Editor editor = sharedPreferences.edit();
editor.remove("fullname");
editor.apply();
```



FPT Education

FPT POLYTECHNIC

Thank you